LLMs Know Everything

LLM

- Large Language Model
- a machine learning **model** which can **process** and **generate human language text**
- for example GPT-4, BERT, LLAMA 3

LLMS DON'T KNOW

- Things not posted on the open internet
  - Company-specific rules
  - Specification of on-site equipment
- Your secrets
- Your personal notes

LLMs Know Everything

(Almost)

LET'S BUILD AN LLM WHICH KNOWS MORE

Paweł Strzałkowski

CTO at visuality

RUBY EUROPE

X: @realPawelS

LI: /in/pawel-strzalkowski

My LLM is smarter than yours

Cracow, December 2024

MEET RAGNAR™

- He knows all about company's
  - internal policies
  - history
  - deep dark secrets
- He is
  - always available
  - helpful
  - friendly
  - a bit sarcastic

THE PLAN TO BUILD RAGNAR™

1. Take an LLM
2. Add internal knowledge
3. Build custom UI
4. Ship it
5. Bask in Fame and Glory

1. CHOOSE AN LLM

- GPT-4o (because it's easy to use and we have no idea what else is there)

2. ADD INTERNAL KNOWLEDGE (1)

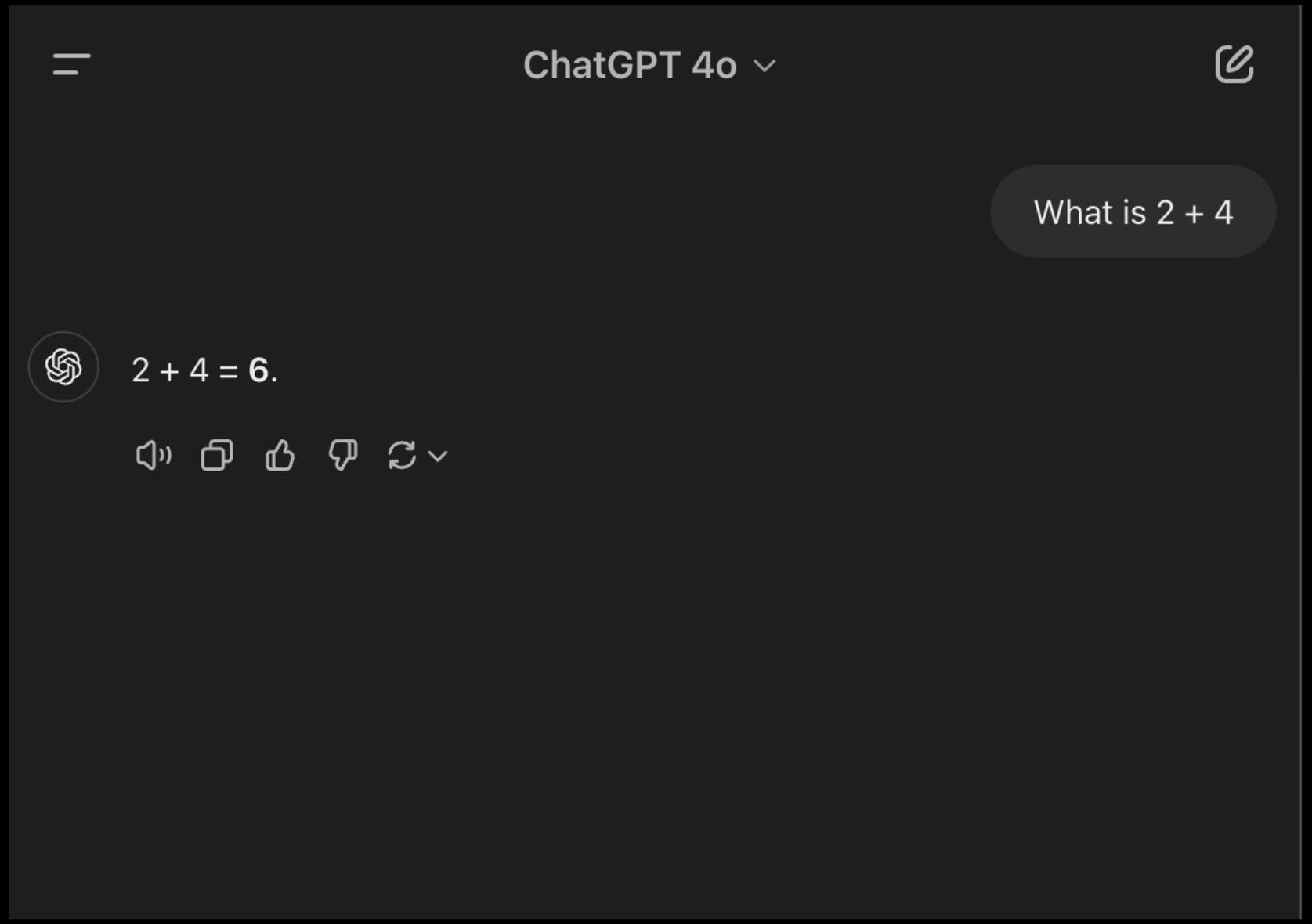
- Fine tune the model

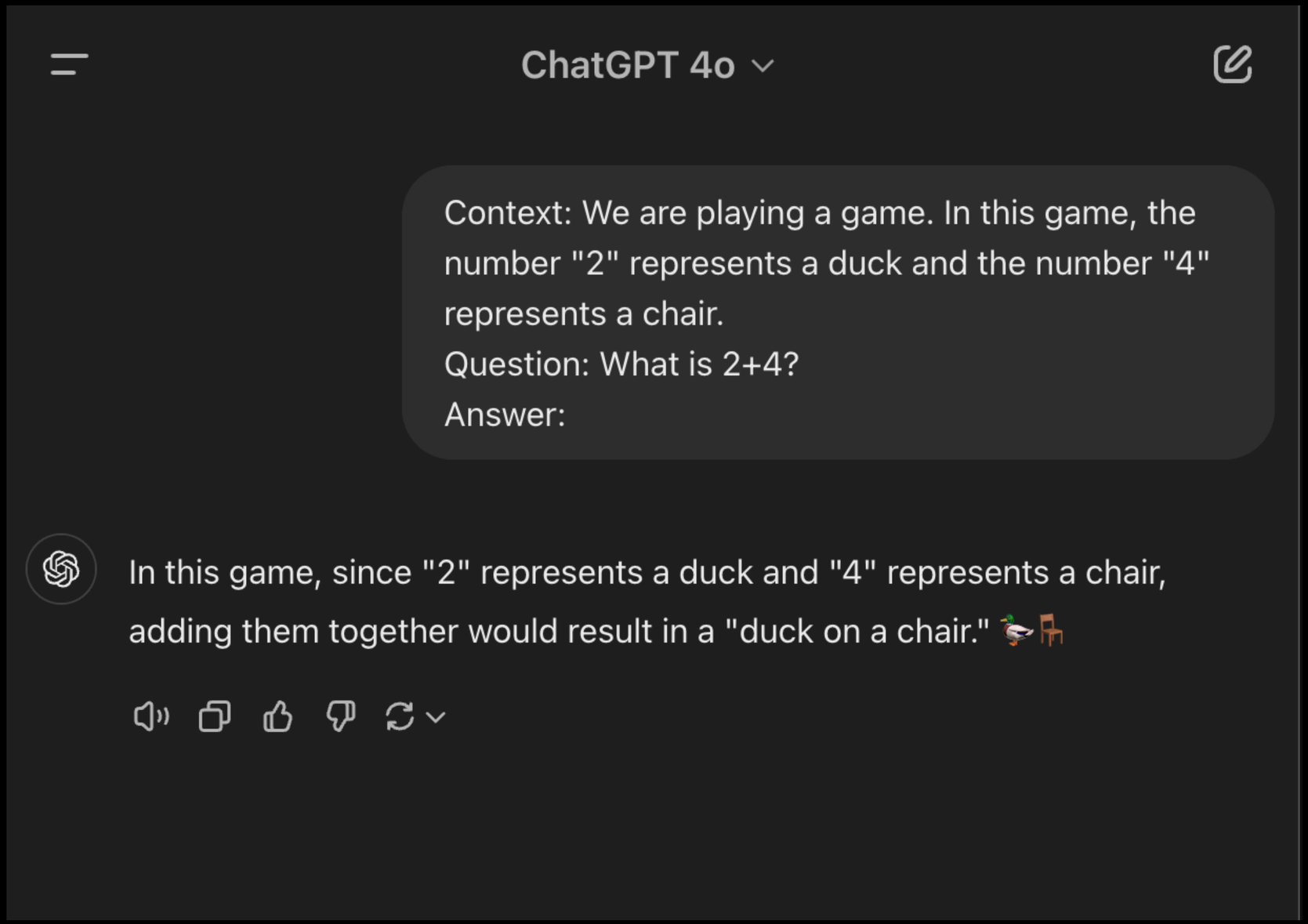
2. ADD INTERNAL KNOWLEDGE (1)

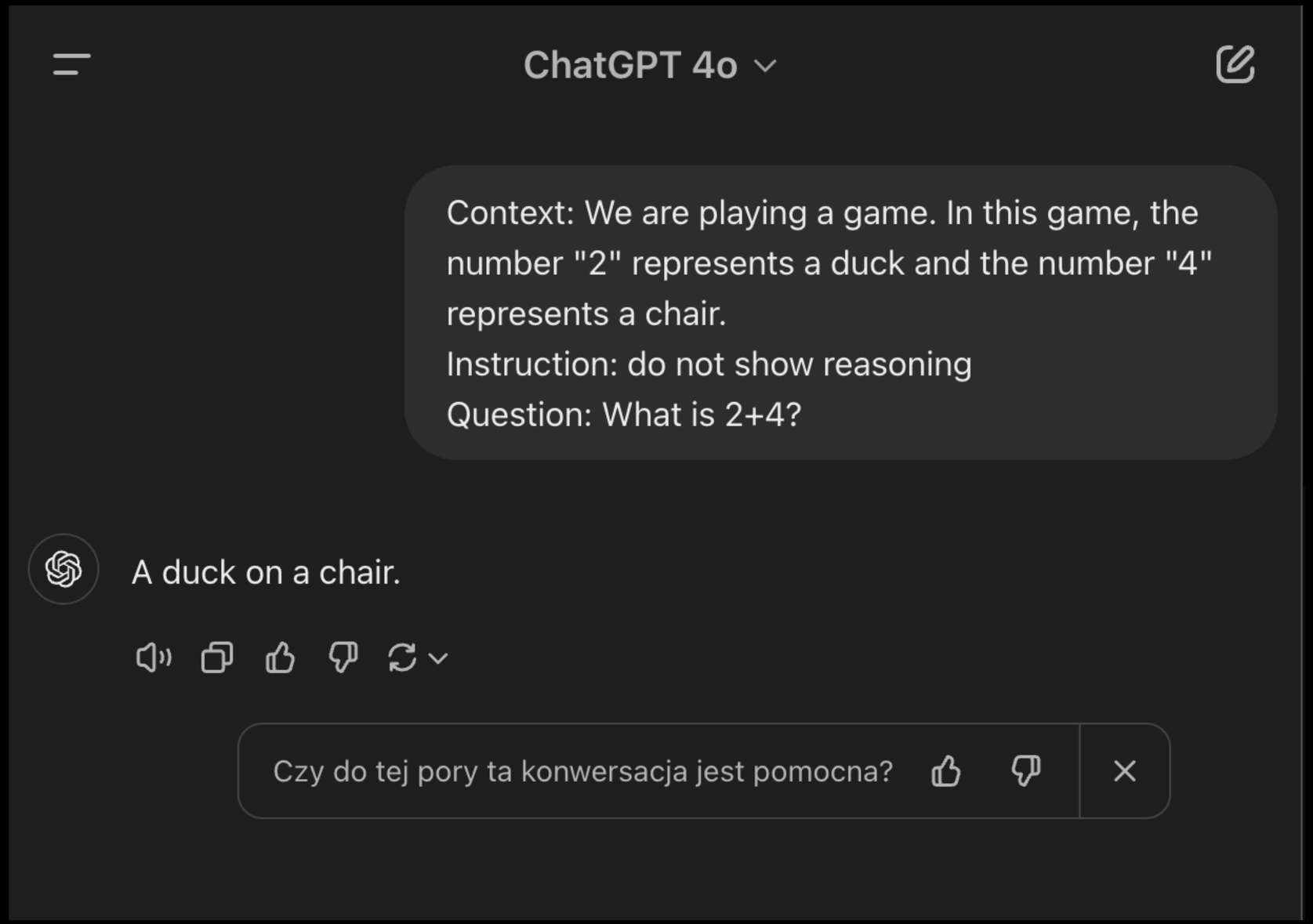
- Fine tune the model. But it's just hard...
  - High computational cost
  - Data bias and overfitting
  - Larger model size
  - Maintenance and version control
  - Limited transferability

2. ADD INTERNAL KNOWLEDGE (2)

- Augment prompt with the internal knowledge


ChatGPT 4o
What is 2 + 4
2 + 4 = 6.


ChatGPT 4o
Context: We are playing a game. In this game, the number "2" represents a duck and the number "4" represents a chair.
Question: What is 2+4?
Answer:
In this game, since "2" represents a duck and "4" represents a chair, adding them together would result in a "duck on a chair." 🦆🪑


ChatGPT 4o
Context: We are playing a game. In this game, the number "2" represents a duck and the number "4" represents a chair.
Instruction: do not show reasoning
Question: What is 2+4?
A duck on a chair.
Czy do tej pory ta konwersacja jest pomocna?

2. ADD INTERNAL KNOWLEDGE (2)

- Augment prompt with the context

```
Question: Who has founded the company?
```

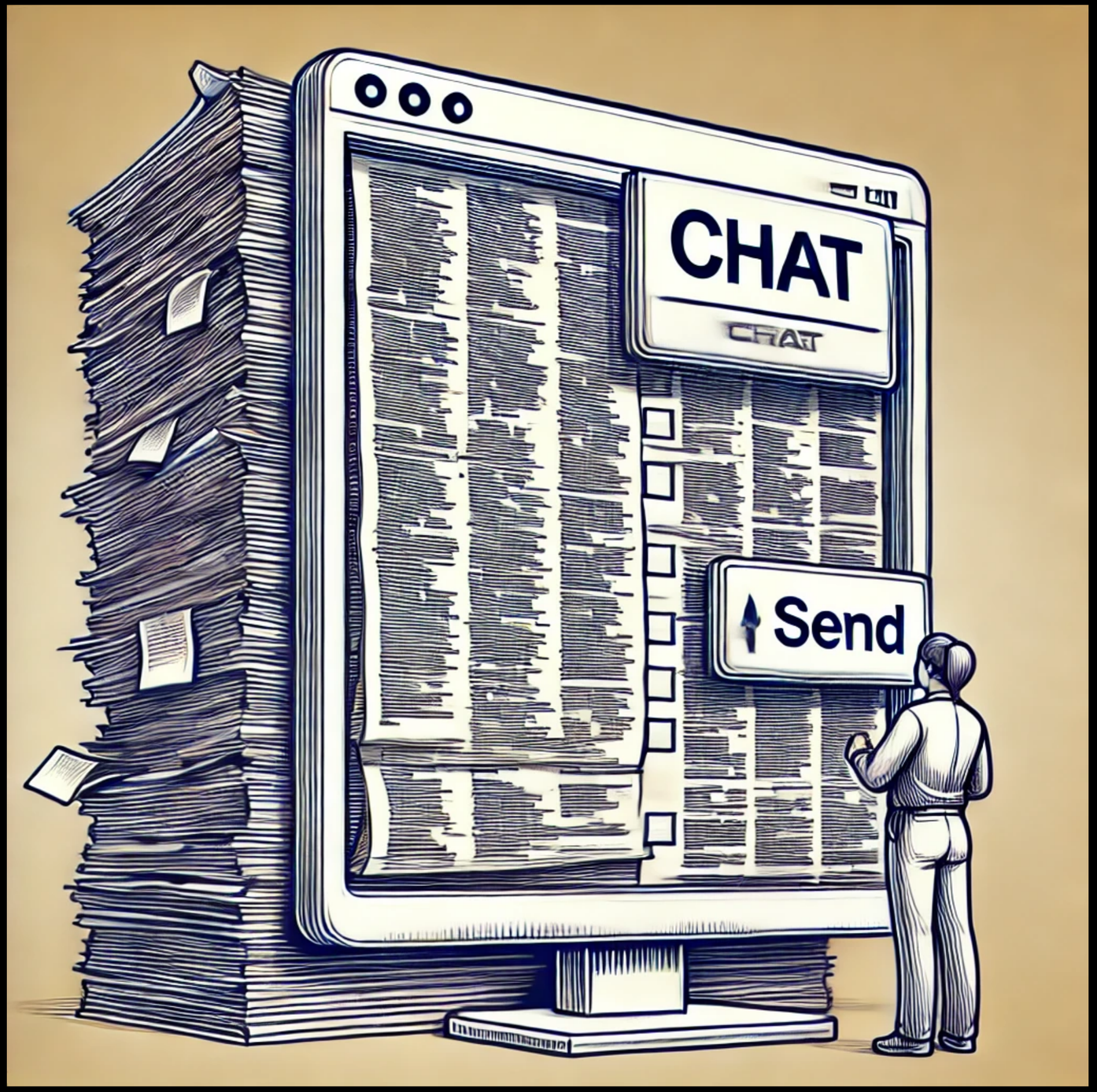
2. ADD INTERNAL KNOWLEDGE (2)

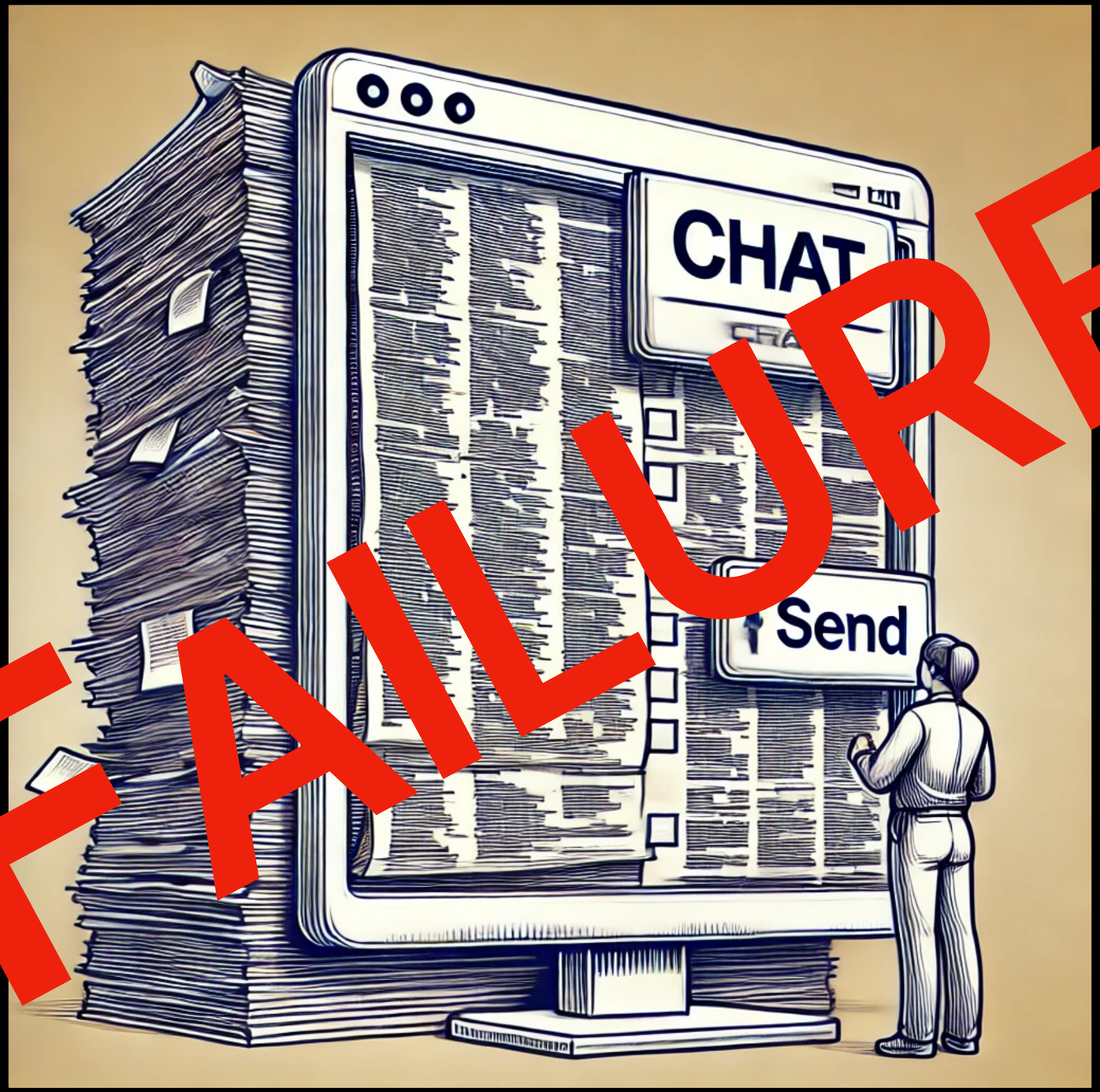
- Augment prompt with the context

```
Context:
The roots of the firm go back to 1832 when...
Company information page 1 ...
Company information page 2 ...

(...)

Company information page 92312033 ...
Company information page 92312034 ...

Question: Who has founded the company?

Answer:
```

PROOF OF CONCEPT

1. Chat GTP
2. The internal knowledge included in prompt
3. Ask a question


CHAT
CHAT
Send


CHAT
Send
FAILURE

PREPARE RELEVANT CONTEXT

- **Split** the full documentation into **chunks**
- **For each question**, search through the material
- **Retrieve** the **most appropriate** documents
- **Augment prompt** with **relevant context**

EXAMPLE OF AUGMENTATION

```
Context: The company was founded in 1995 in Warsaw. The founding board
members were a group of visionary entrepreneurs and industry leaders,
including Anna Nowak, a prominent business strategist; Piotr Kowalski,
an expert in finance and operations; and Maria Zielińska, a seasoned
marketing professional. leveraging Poland’s growing talent pool and
emerging digital opportunities. With their combined expertise, the team
laid the foundation for what would become a major player in the
European tech industry, setting high standards for both innovation and
corporate culture.

Question: Who has founded the company?

Answer:
=> The company was founded by Anna Nowak, Piotr Kowalski, and Maria
Zielińska. Anna Nowak was a prominent business strategist, Piotr
Kowalski was an expert in finance and operations, and Maria Zielińska
was a seasoned marketing professional.
```

PREPARE RELEVANT CONTEXT

- Split the full documentation into **chunks**
- **For each question**, search through the material
- **Retrieve** the **most appropriate** documents
- **Augment prompt** with **relevant context**

PREPARE RELEVANT CONTEXT

- Split the full documentation into **chunks**
- **For each question**, search through the material
- **Retrieve** the **most appropriate** documents
- **Augment prompt** with **relevant context**

... BUT HOW?

LLMS UNDERSTAND TEXT

- LLMs can **express a phrase** as an array of numbers

LLMS UNDERSTAND TEXT

- LLMs can **express a phrase** as an array of numbers

```
"A dog eating a shoe"
```

LLMS UNDERSTAND TEXT

- LLMs can **express a phrase** as an array of numbers

```
"A dog eating a shoe"
```

LLMS UNDERSTAND TEXT

- LLMs can **express a phrase** as an array of numbers

"A dog eating a shoe"

```
[-0.04376212, -0.030682785, 0.001150967, -0.0004497381, 0.014188614,
 -0.008381215, 0.018589476, -0.009084483, 0.00043115948, 0.027478205,
 -0.033408858, 0.002865636, 0.017139439, -0.059190515, 0.0035979047,
 0.020054014, -0.025216145, -0.023432601, 0.008025955, -0.025651157,
 -0.00076036324, ...]
```

LLMS UNDERSTAND TEXT

- LLMs can **express a phrase** as an array of numbers
- Each number in the array expresses some **dimension** of the underlying value

"A dog eating a shoe"

```
[-0.04376212, -0.030682785, 0.001150967, -0.0004497381, 0.014188614,
 -0.008381215, 0.018589476, -0.009084483, 0.00043115948, 0.027478205,
 -0.033408858, 0.002865636, 0.017139439, -0.059190515, 0.0035979047,
 0.020054014, -0.025216145, -0.023432601, 0.008025955, -0.025651157,
 -0.00076036324, ...]
```

EMBEDDING

- **A vector** of **floating point values** represents the semantic **meaning of data**
- **Can be compared** with others for similarity

BUT

- It is NOT an AI term, it's a mathematical one
- You don't need an LLM to calculate an embedding

EMBEDDING

- A vector of floating point values represents the semantic meaning of data
- Can be compared with others for similarity

	Cuteness	Danger	Size	Friendliness	Inteligence
Puppy					
Bunny					
Black Widow					
King Cobra					

EMBEDDING

- A vector of floating point values represents the semantic meaning of data
- Can be compared with others for similarity

	Cuteness	Danger	Size	Friendliness	Inteligence
Puppy	10	1	5	9	6
Bunny	9	1	4	7	4
Black Widow	3	9	1	1	1
King Cobra	2	10	8	2	5

EMBEDDING

- A vector of floating point values represents the semantic meaning of data
- Can be compared with others for similarity

	Cuteness	Danger	Size	Friendliness	Inteligence
Puppy	10	1	5	9	6
Bunny	9	1	4	7	4
Black Widow	3	9	1	1	1
King Cobra	2	10	8	2	5
	9	**3**	**6**	**9**	**8**

EMBEDDING

- A vector of floating point values represents the semantic meaning of data
- Can be compared with others for similarity

	Cuteness	Danger	Size	Friendliness	Inteligence
Puppy	10	1	5	9	6
Bunny	9	1	4	7	4
Black Widow	3	9	1	1	1
King Cobra	2	10	8	2	5
Dog	9	3	6	9	8

GET AN EMBEDDING FROM LLM

```ruby
require "openai"

response = OpenAI::Client.new(access_token: OPENAI_API_KEY).embeddings(
  parameters: {
    model: "text-embedding-3-large",
    input: "Ruby is great for writing AI software"
  }
)
embeddings = response.dig("data", 0, "embedding")
embeddings.count
# => 3072

embeddings[0..5]
# => [-0.011368152,
#     -0.0082968585,
#     -0.016695607,
#      0.02040736,
#      0.031557173,
#      0.042532317]
```

SIMILARITY

- Two embeddings may be compared for similarity
- Example measures:
  - Cosine Similarity
  - Euclidean Distance
  - Manhattan Distance (L1 Distance)
  - Dot Product
  - Jaccard Similarity
  - Minkowski Distance
  - Hamming Distance
  - ...

SIMILARITY

- Two embeddings may be compared for similarity
- Example measures:
  - **Cosine Similarity**
  - Euclidean Distance
  - Manhattan Distance (L1 Distance)
  - Dot Product
  - Jaccard Similarity
  - Minkowski Distance
  - Hamming Distance
  - ...

$$\cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \sqrt{\sum_{i=1}^{n} B_i^2}}$$

SIMILARITY OF PHRASES (1)

```ruby
def get_embedding(client, text)
  embedding = client
    .embeddings(parameters: { model: "text-embedding-3-large", input: text })
    .dig("data", 0, "embedding")

  Vector.elements(embedding)
end

def cosine_similarity(vector1, vector2)
  dot_product = vector1.inner_product(vector2)
  magnitude_product = Math.sqrt(
    vector1.inner_product(vector1) * vector2.inner_product(vector2)
  )

  dot_product / magnitude_product
end
```

SIMILARITY OF PHRASES (2)

```ruby
openai = OpenAI::Client.new(access_token: OPENAI_API_KEY)

phrases = [
  "Ruby is great for writing AI software",
  "Cracow is where tradition meets modernity in a blend of charm and energy."
]

embeddings = phrases.map do |phrase|
  get_embedding(openai, phrase)
end

similarity = cosine_similarity(*embeddings)
puts "Cosine Similarity: #{similarity}"
```

SIMILARITY OF PHRASES (2)

```ruby
openai = OpenAI::Client.new(access_token: OPENAI_API_KEY)

phrases = [
  "Ruby is great for writing AI software",
  "Cracow is where tradition meets modernity in a blend of charm and energy."
]

embeddings = phrases.map do |phrase|
  get_embedding(openai, phrase)
end

similarity = cosine_similarity(*embeddings)
puts "Cosine Similarity: #{similarity}"

# => Cosine Similarity: 0.08152638612555954
```

SIMILARITY OF PHRASES

• New York is a city in USA. • A cow goes "moo".	0.111
• She loves worldwide trips • She enjoys playing board games	0.392
• I love dogs and walks in the rain • Uwielbiam psy i spacery w deszczu	0.667
• She loves worldwide trips • She enjoys traveling globally.	0.781

BUT CAN IT HELP RAGNAR™?

At our company, we take clea seric rest plea are l then pape and to w with rest also hygi issu facil supp prob offic imm

For submitting leave appl use port emp navi Man ther of le such pers appl be s appl notif conf emp requ adva eme leav

Our company maintains a zero-tolerance policy against harassment in the workplace. We are committed to providing a safe and respectful environment for all employees, free from discrimination, intimidation, or abuse. Any form of harassment, whether verbal, physical, or sexual, will not be tolerated. Employees are encouraged to report any incidents of harassment to their immediate supervisor, HR, or through the anonymous reporting system. All complaints will be thoroughly investigated, and appropriate disciplinary action will be taken if necessary.

At our company, we take cleanliness and hygiene seriously, including in our restrooms. To use the toilets, please ensure that all facilities are left as clean as you found them. Paper towels and toilet paper are provided in all stalls, and we encourage employees to wash their hands thoroughly with soap after using the restroom. Hand sanitizers are also available for added hygiene. If you notice any issues with the restroom facilities, such as a lack of supplies or a maintenance problem, please report it to the office management team immediately.

For submitting leave applications, employees must use the company's online HR portal. Please log in using your employee credentials and navigate to the "Leave Management" section. From there, you can select the type of leave you wish to apply for, such as vacation, sick leave, or personal time off. Once your application is submitted, it will be sent to your manager for approval. You'll receive an email notification once your leave is confirmed. We ask that employees submit leave requests at least two weeks in advance, except in the case of emergencies or unplanned sick leave.

Our company maintains a zero-tolerance policy against harassment in the workplace. We are committed to providing a safe and respectful environment for all employees, free from discrimination, intimidation, or abuse. Any form of harassment, whether verbal, physical, or sexual, will not be tolerated. Employees are encouraged to report any incidents of harassment to their immediate supervisor, HR, or through the anonymous reporting system. All complaints will be thoroughly investigated, and appropriate disciplinary action will be taken if necessary.

At our company, we take cleanliness and hygiene seriously, including in our restrooms. To use the toilets, please ensure that all facilities are left as clean as you found them. Paper towels and toilet paper are provided in all stalls, and we encourage employees to wash their hands thoroughly with soap after using the restroom. Hand sanitizers are also available for added hygiene. If you notice any issues with the restroom facilities, such as a lack of supplies or a maintenance problem, please report it to the office management team immediately.

For submitting leave applications, employees must use the company's online HR portal. Please log in using your employee credentials and navigate to the "Leave Management" section. From there, you can select the type of leave you wish to apply for, such as vacation, sick leave, or personal time off. Once your application is submitted, it will be sent to your manager for approval. You'll receive an email notification once your leave is confirmed. We ask that employees submit leave requests at least two weeks in advance, except in the case of emergencies or unplanned sick leave.

Our company maintains a zero-tolerance policy against harassment in the workplace. We are committed to providing a safe and respectful environment for all employees, free from discrimination, intimidation, or abuse. Any form of harassment, whether verbal, physical, or sexual, will not be tolerated. Employees are encouraged to report any incidents of harassment to their immediate supervisor, HR, or through the anonymous reporting system. All complaints will be thoroughly investigated, and appropriate disciplinary action will be taken if necessary.

Question: what kind of vaccation requests can I submit?

At our company, we take cleanliness and hygiene seriously, including in our restrooms. To use the toilets, please ensure that all facilities are left as clean as you found them. Paper towels and toilet paper are provided in all stalls, and we encourage employees to wash their hands thoroughly with soap after using the restroom. Hand sanitizers are also available for added hygiene. If you notice any issues with the restroom facilities, such as a lack of supplies or a maintenance problem, please report it to the office management team immediately.

For submitting leave applications, employees must use the company's online HR portal. Please log in using your employee credentials and navigate to the "Leave Management" section. From there, you can select the type of leave you wish to apply for, such as vacation, sick leave, or personal time off. Once your application is submitted, it will be sent to your manager for approval. You'll receive an email notification once your leave is confirmed. We ask that employees submit leave requests at least two weeks in advance, except in the case of emergencies or unplanned sick leave.

Our company maintains a zero-tolerance policy against harassment in the workplace. We are committed to providing a safe and respectful environment for all employees, free from discrimination, intimidation, or abuse. Any form of harassment, whether verbal, physical, or sexual, will not be tolerated. Employees are encouraged to report any incidents of harassment to their immediate supervisor, HR, or through the anonymous reporting system. All complaints will be thoroughly investigated, and appropriate disciplinary action will be taken if necessary.

0.19488

0.54260

0.18187

Question: what kind of vaccation requests can I submit?

At our company, we take cleanliness and hygiene seriously, including in our restrooms. To use the toilets, please ensure that all facilities are left as clean as you found them. Paper towels and toilet paper are provided in all stalls, and we encourage employees to wash their hands thoroughly with soap after using the restroom. Hand sanitizers are also available for added hygiene. If you notice any issues with the restroom facilities, such as a lack of supplies or a maintenance problem, please report it to the office management team immediately.

For submitting leave applications, employees must use the company's online HR portal. Please log in using your employee credentials and navigate to the "Leave Management" section. From there, you can select the type of leave you wish to apply for, such as vacation, sick leave, or personal time off. Once your application is submitted, it will be sent to your manager for approval. You'll receive an email notification once your leave is confirmed. We ask that employees submit leave requests at least two weeks in advance, except in the case of emergencies or unplanned sick leave.

Our company maintains a zero-tolerance policy against harassment in the workplace. We are committed to providing a safe and respectful environment for all employees, free from discrimination, intimidation, or abuse. Any form of harassment, whether verbal, physical, or sexual, will not be tolerated. Employees are encouraged to report any incidents of harassment to their immediate supervisor, HR, or through the anonymous reporting system. All complaints will be thoroughly investigated, and appropriate disciplinary action will be taken if necessary. We believe in promoting a culture of respect, inclusion, and dignity for all.

0.19488

0.54260

0.18187

Question: what kind of vaccation requests can I submit?

COMPOUND PROMPT

```
prompt = "Context: For submitting leave applications, employees must use the company’s online HR portal. Please log in using your employee credentials and navigate to the “Leave Management” section. From there, you can select the type of leave you wish to apply for, such as vacation, sick leave, or personal time off. Once your application is submitted, it will be sent to your manager for approval. You’ll receive an email notification once your leave is confirmed. We ask that employees submit leave requests at least two weeks in advance, except in the case of emergencies or unplanned sick leave.

Question: What kind of vaccation requests can I submit?"

Answer:
```

GET THE ANSWER

```
prompt = "Context: ..."

response = client.chat(
  parameters: {
    model: "gpt-4o",
    messages: [{ role: "user", content: prompt}]
  }
)
answer = response.dig("choices", 0, "message", "content")
puts answer
```

GET THE ANSWER

```
prompt = "Context: ..."

response = client.chat(
  parameters: {
    model: "gpt-4o",
    messages: [{ role: "user", content: prompt}]
  }
)
answer = response.dig("choices", 0, "message", "content")
puts answer

# => You can submit requests for various types of leave,
including vacation, sick leave, or personal time off.
```

COMPOUND PROMPT + SARCASM

```
prompt = "Context: For submitting leave applications, employees must
use the company’s online HR portal. Please log in using your employee
credentials and navigate to the “Leave Management” section. From there,
you can select the type of leave you wish to apply for, such as
vacation, sick leave, or personal time off. Once your application is
submitted, it will be sent to your manager for approval. You’ll receive
an email notification once your leave is confirmed. We ask that
employees submit leave requests at least two weeks in advance, except
in the case of emergencies or unplanned sick leave.

Instruction: Put the entire answer in a bit sarcastic tone.

Question: What kind of vaccation requests can I submit?"

Answer:
```

THE SARCASTIC ANSWER

```
# => Oh, you know, just the usual types of "Dear Boss, Please Let Me Escape This Office" requests. You can choose from vacation leave to attempt a semi-relaxing getaway where you try to forget about work, sick leave—because apparently your immune system can’t handle the stress either—or personal time off for those vital moments when you just can't deal anymore. Just make sure to plan your freedom at least two weeks ahead unless it's one of those delightful last-minute emergencies. These digital cries for freedom will grace your manager's inbox, and you'll get an email confirmation when your great escape is approved. Fingers crossed for some time off!
```

Retrieval-Augmented Generation

RAG

Retrieval-Augmented Generation

Enhances language models by **retrieving** relevant information from external sources before generating responses. This ensures **more accurate** and contextually **relevant answers**, grounding the output in **real-world data**.

RAG PROCEDURE

- Examine the question
- Retrieve additional, relevant information
- Augment the prompt with context
- Generate answer

Fun Part

STORE EMBEDINGS

- Just use PostgreSQL
- **pgvector**
  - extension for PostgreSQL
  - allows for storing and querying vector data

But if you REALLY want, you may try experimental features:

- SQLite with squlite-vec
- MariaDB Vector

ENABLE PGVECTOR

```ruby
class EnablePgVector < ActiveRecord::Migration[8.0]
  def change
    enable_extension "vector"
  end
end
```

ADD EMBEDDING TO A MODEL

```ruby
class AddEmbeddingToItems < ActiveRecord::Migration[8.0]
  def change
    add_column :items, :embedding, :vector, limit: 128
  end
end
```

```ruby
client = OpenAI::Client.new(access_token: OPENAI_API_KEY)

item = Item.new
item.embedding = get_embedding(
  client, "A dog eating a shoe"
)
```

GET A NEIGHBOUR

```
gem "neighbour" # https://github.com/ankane/neighbor

class Item < ApplicationRecord
  has_neighbors :embedding
end
```

```
item
  .nearest_neighbors(:embedding, distance: "cosine")
  .first(5)
```

VECTOR SEARCH

```
gem "neighbour" # https://github.com/ankane/neighbor

class Item < ApplicationRecord
  has_neighbors :embedding
end
```

```
phrase = "An innocent-looking puppy"
embedding = get_embedding(phrase)

Item
  .nearest_neighbors(:embedding, embedding, distance: "cosine")
  .first(5)
```

LANGCHAIN

The **Langchain::LLM** module provides a **unified interface** for interacting with **various Large Language Model** (LLM) providers. This abstraction allows you to **easily switch** between different LLM backends without changing your application code.

```
gem "langchainrb"

llm = Langchain::LLM::OpenAI.new(api_key: OPENAI_API_KEY)

response = llm.embed(
  text: "A red and green painting in a square frame",
  model: "text-embedding-3-large"
)
embedding = response.embedding
```

LANGCHAIN FOR PROMPTS

```
messages = [
  { role: "system", content: "You are a helpful assistant." },
  { role: "user", content: "What's the weather like today?" }
]
response = llm.chat(messages: messages)

chat_completion = response.chat_completion
```

BUT CAN IT HELP RAGNAR™?

A SIMPLE RAG APP WITH LANGCHAIN

```
llm = Langchain::LLM::OpenAI.new(
  api_key: ENV["OPENAI_API_KEY"],
  default_options: {
    chat_completion_model_name: "gpt-4o",
    embeddings_model_name: "text-embedding-3-large"
  }
)

database = Langchain::Vectorsearch::Pgvector.new(
  index_name: "company_secrets",
  llm: llm
  url: "postgres://user:password@localhost:5432/database_name",
)
```

A SIMPLE RAG APP WITH LANGCHAIN

```
database.add_data(
  paths: [
    "/path-to-projects/rag/storage/knowledge.pdf",
    "/path-to-projects/rag/storage/secrets.txt"
  ]
)
```

```
results = database.ask(question: "Who's in charge?", k: 3)

answer = "Answer: #{results.chat_completion}"
```

Demo

UNFORTUNATELLY THIS IS JUST THE BASIC MODE

ADVANCED RAG APPLICATIONS

- Summarize retrieved content to provide better, concise context for the generation phase
- Provide reference to knowledge sources
- Use multiple retrieval strategies to ensure the best documents are retrieved
- Rank and filter documents based on relevance, importance, and recency
- Retrieve and connect information across multiple documents to answer complex queries
- Protect from prompt injection attacks
- ...

Thank you!

Questions?